

Enterprise Open Source: Delivering Value versus Cutting Costs

Gartner Open Source Summit,
28 September, 2006
Phoenix, AZ

Stephen Walli
CTO and VP Engineering,
Descant, Inc.

What this talk is about

- If you only take away one thing from this talk, it's to understand that you already know how to apply open source software in your organizations
- It is no different than any other technology decision you make -- there's nothing inherently "new" here
- There isn't any mystery to the TCO or ROI around open source software -- the rules haven't changed, but the yardstick might be different

Free and Open Source Software



Source code for the software is always available, and the user of the software is licensed (always) to change and redistribute the software without fee, penalty or asking permission

Copyright, Stephen R. Walli, some rights reserved

Distributed under a Creative Commons, Attribution-NonCommercial-NoDerivs 2.0 Deed

Barriers for Using Open Source Software in the Enterprise

Concerns about support

92%

Insufficient knowledge about open source alternatives

74%

Insufficient knowledge about open source business value

69%

Concerns about IP, licensing and legal issues

65%

Lack of internal expertise how to migrate to open source

64%

Lack of internal expertise for installing/customizing open source

56%

Concerns that open source lacks needed features & functionality

54%

Why aren't you using OSS?

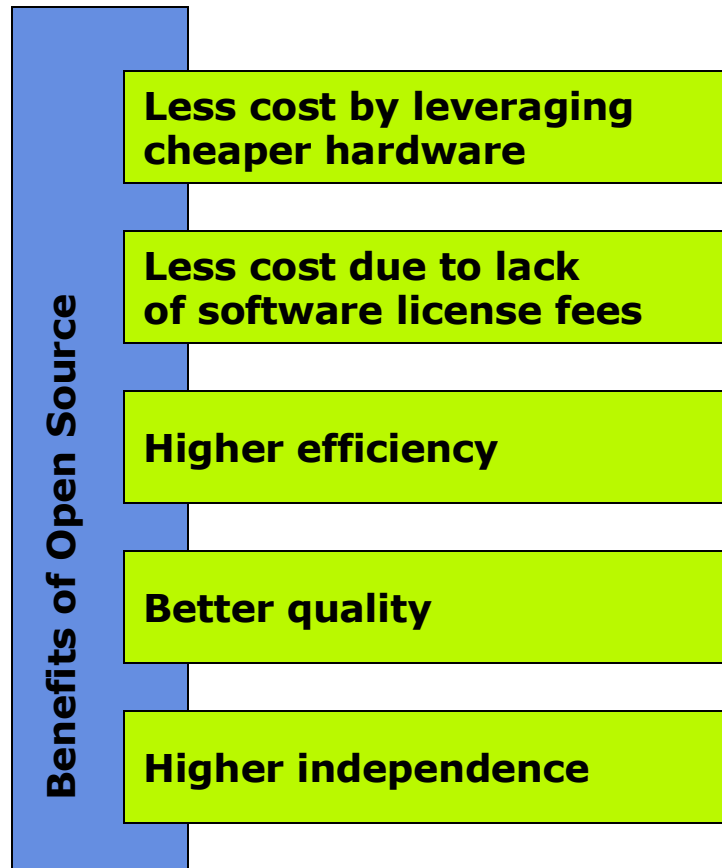
* Companies with > USD 1 billion revenue

Source: Optaros/InformationWeek Survey 2005

Copyright, Stephen R. Walli, some rights reserved

Distributed under a Creative Commons, Attribution-NonCommercial-NoDerivs 2.0 Deed

The Benefits of Using Open Source Software and Components



Reasons

- Usage of commodity hardware (TCO)
- Better leverage of hardware resources (ROA)
- No software licence costs (TCO)
- Very competitive software maintenance and support market (TCO)
- Availability of skills and people (Cost)
- Platform for development (Value)
- Collaborative software development process (Cost/Value)
- Full access to software code (Value)
- No vendor lock in
- Non monopolistic approach
- Standard compliance

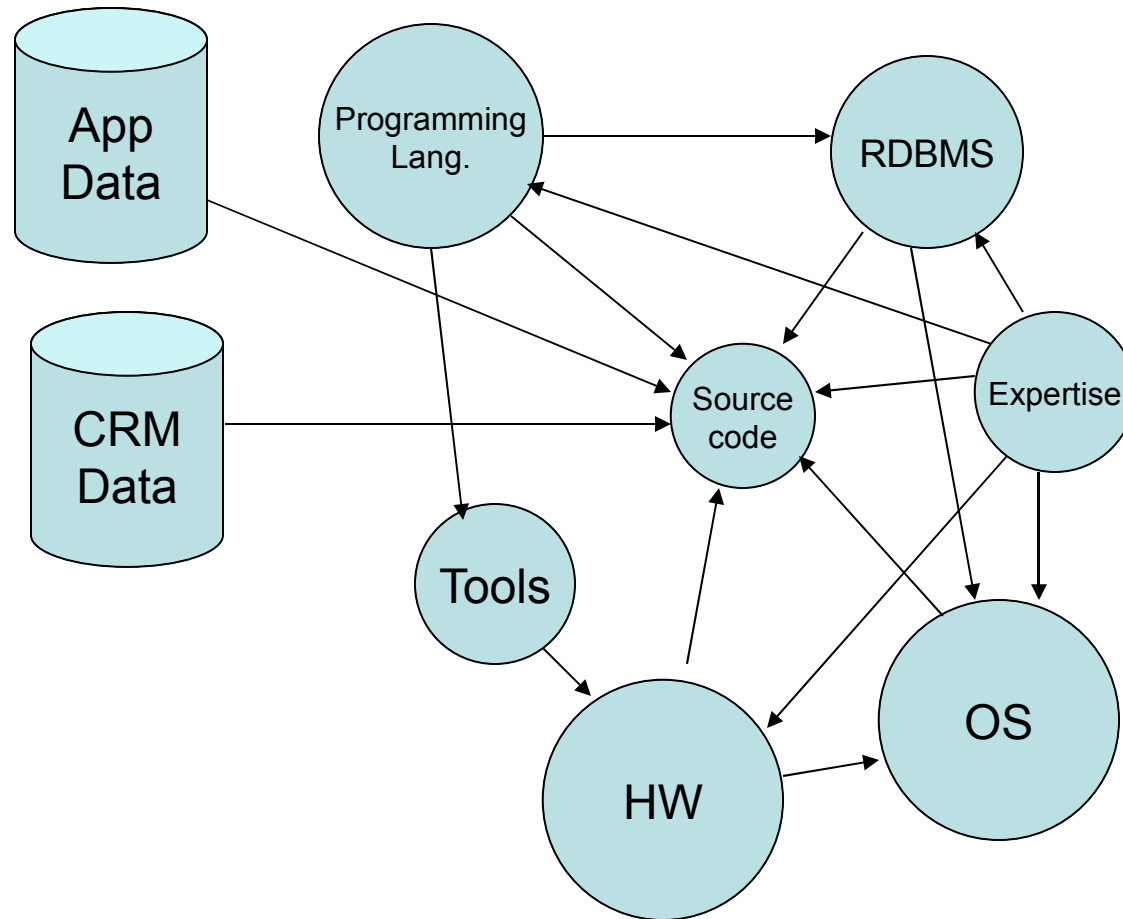
Return on Investment ...is bogus?

- Return on investment (ROI) is no longer seen as a key indicator of what does and what does not make technology investments work, a poll of 400 IT and financial directors in the UK has shown
 - http://www.theregister.co.uk/2002/07/09/roi_faulty_as_measurement/
- Brian Hadfield Managing Director of Unisys UK told ComputerWire that ROI is best viewed as a "trailing indicator"
- Executives polled for the survey see value in the ability of IT to increase customer satisfaction, improve information access or allow better knowledge sharing
- Innovation, improved revenue and reduced costs were not seen as being closely tied to the IT function, turning on its head the belief that technology is best sold on the promise of reduced costs or improved revenue

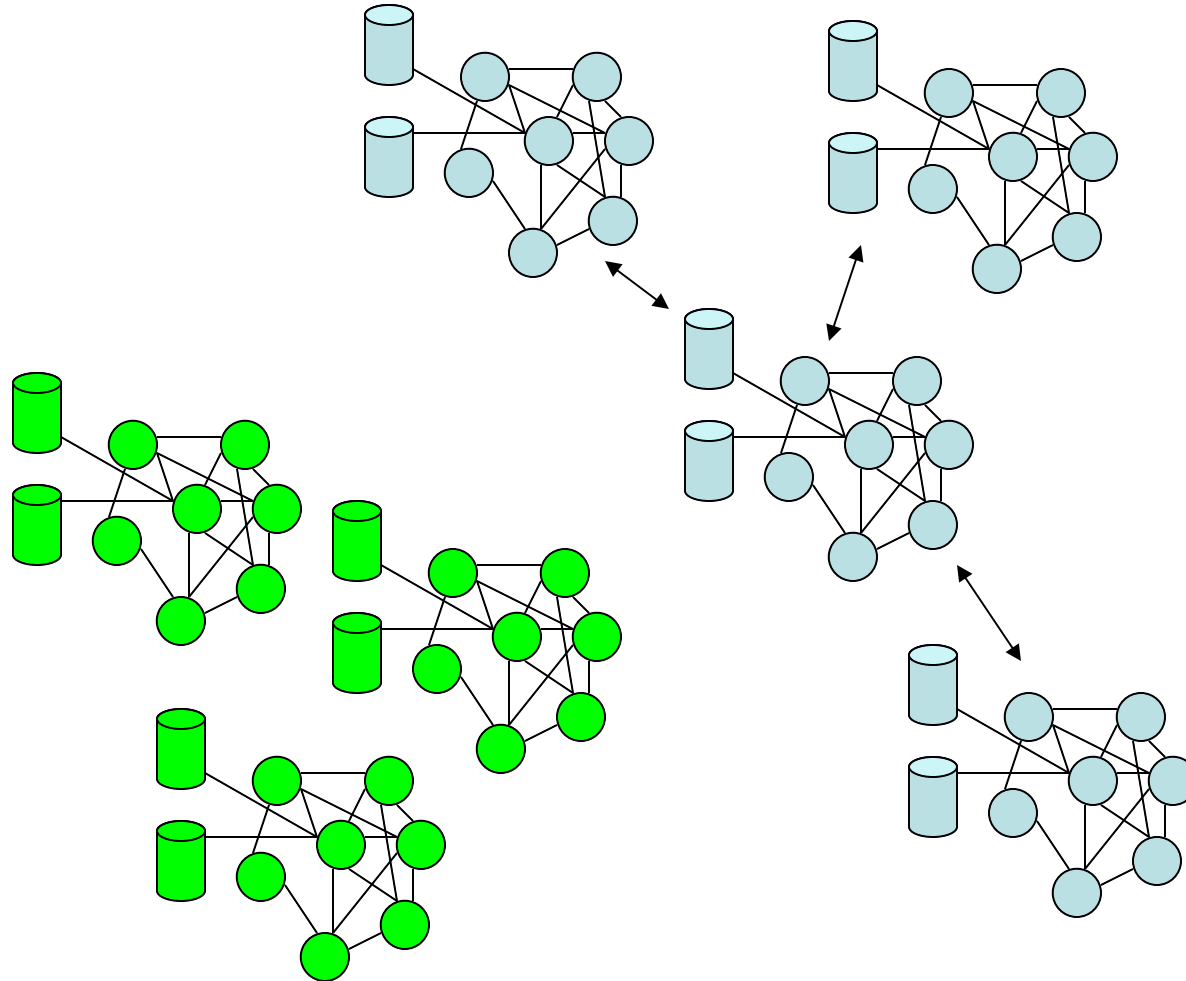
Things to Remember about Open Source

- It's not a stack -- It's a network
 - It's just software
 - It's just economics
 - It's just business
 - It's just software licensing
-
- There's nothing inherently "new" about developing good software, communities, collaboration, or software licensing

The application is actually a network ...



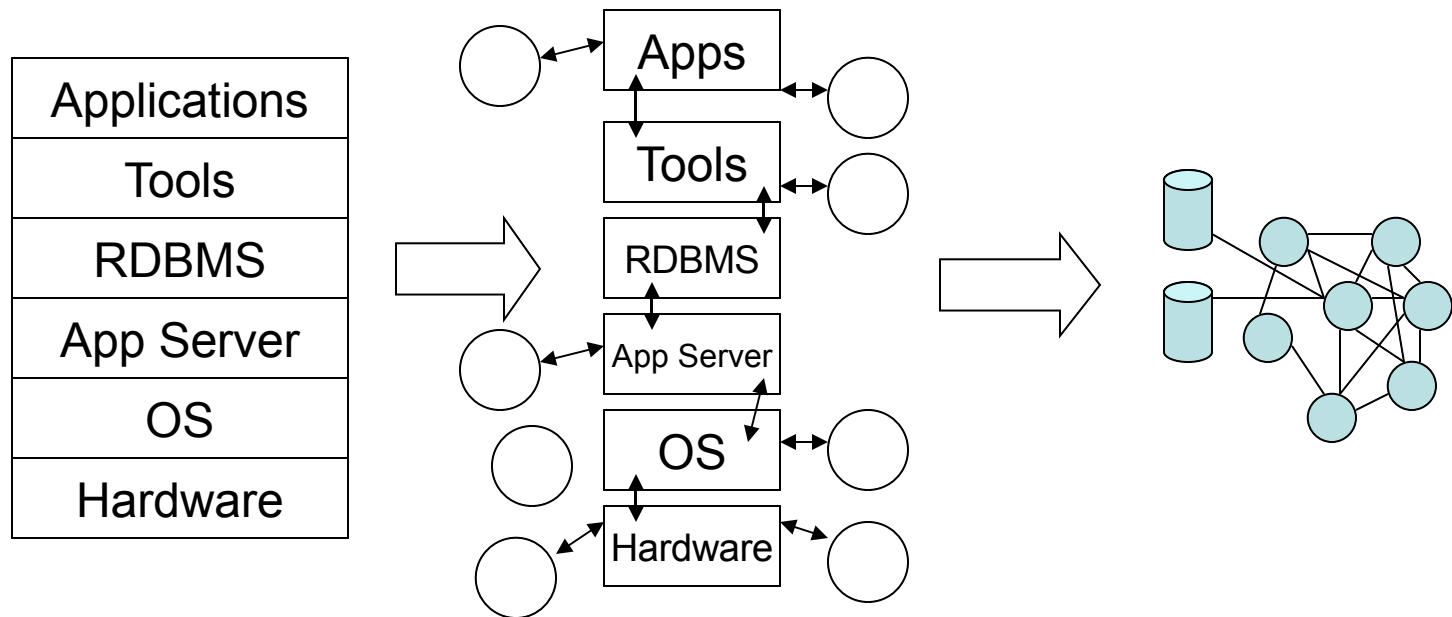
... and the network isn't "simple".



Copyright, Stephen R. Walli, some rights reserved

Distributed under a Creative Commons, Attribution-NonCommercial-NoDerivs 2.0 Deed

It's not a stack – it's a network.



i.e. the “stack” is a view through the network.

It's Just Software

- Good software is developed by good software developers regardless of licensing strategy
- FOSS projects start when a developer writes software and is willing to share it out under a FOSS license
- FOSS projects are not products – they are interesting buckets of technology
- FOSS architecture tends to reflect a UNIX heritage of loosely couple components with stable well defined interfaces

It's Just Economics

- People value their skill sets differently in different contexts
- No one is “working for free” in a real economic sense
- You get more than you give
- If you want to influence you have to participate and contribute
- Companies gain the same benefit as individuals

It's Still Just Economics

- *“A community is just a group of people that share a common interest -- they don't have to like each other.”*
-- Bob Young, former CEO, Red Hat
- A project is an interesting bucket of technology with users and contributors, i.e. the project's community

Projects to Products

- *“The early community is willing to trade time to save money; the late community is willing to trade money to save time. My customer is in the late community.” -- Marten Mickos, CEO, MySQL AB*
- A product is packaged, installable, tested, documented, supported, and maintained for customers
- Companies build products as part of their value proposition to their customer

Engineering Economics

- Two ratios are inescapable in programming:
 - The average number of lines-of-code a developer writes per day
 - The average number of bugs per thousand lines of code
- So it all boils down to writing less code
- Good FOSS projects are the ultimate reuse strategy -- good engineering practice is to design to the scarcest resource -- time!
- It's okay to do custom development again

Business Economics

- Two other apparently inescapable ratios:
 - The maintenance cost over an application's life time (~75%)
 - The number of applications that never get deployed (~60%)
- This is why packaged software took off in the '90s
- This is why packaged software is beginning to fail in certain vertical markets

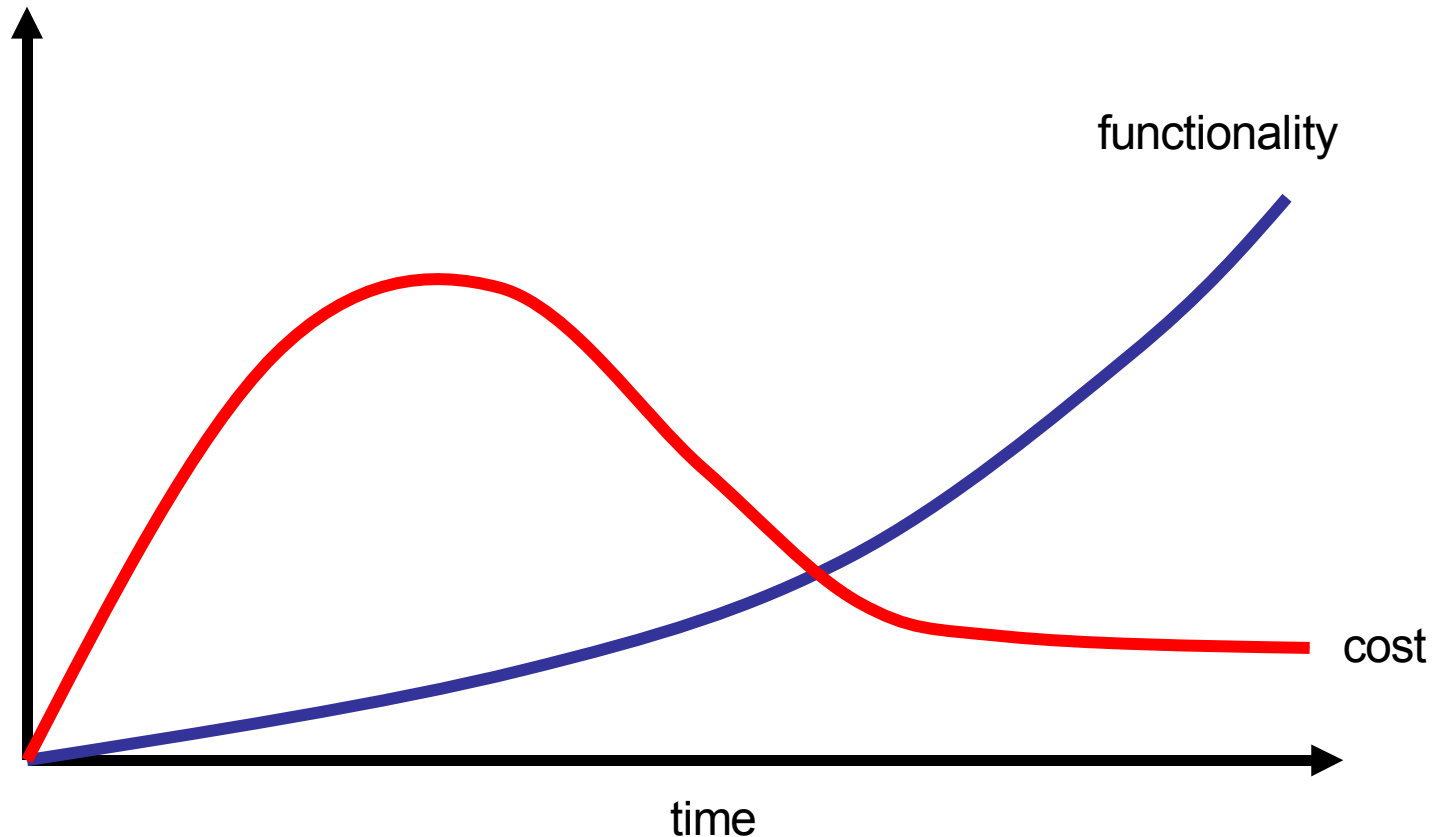
“Using” Open Source Software

- It’s “free” as in beer
- It’s “free” as in freedom
- Community support works
- Self support works
- You can buy professional support from a wealth of players big and small, broad and specialized
- You can rapidly prototype solutions, and they’re “cheap” to throw away, and cheap to deploy

“Buying” Open Source Software

- Buy it just like any other software
- The vendor has profitable business based on different margins
- While you may not “buy” the software -- you do buy the product
- Buy (competitive) support and maintenance with your commercial open source product

“Making” Open Source Software



Share the costs of value creation, maintenance, and support

Copyright, Stephen R. Walli, some rights reserved

Distributed under a Creative Commons, Attribution-NonCommercial-NoDerivs 2.0 Deed

IT As a Business Driver

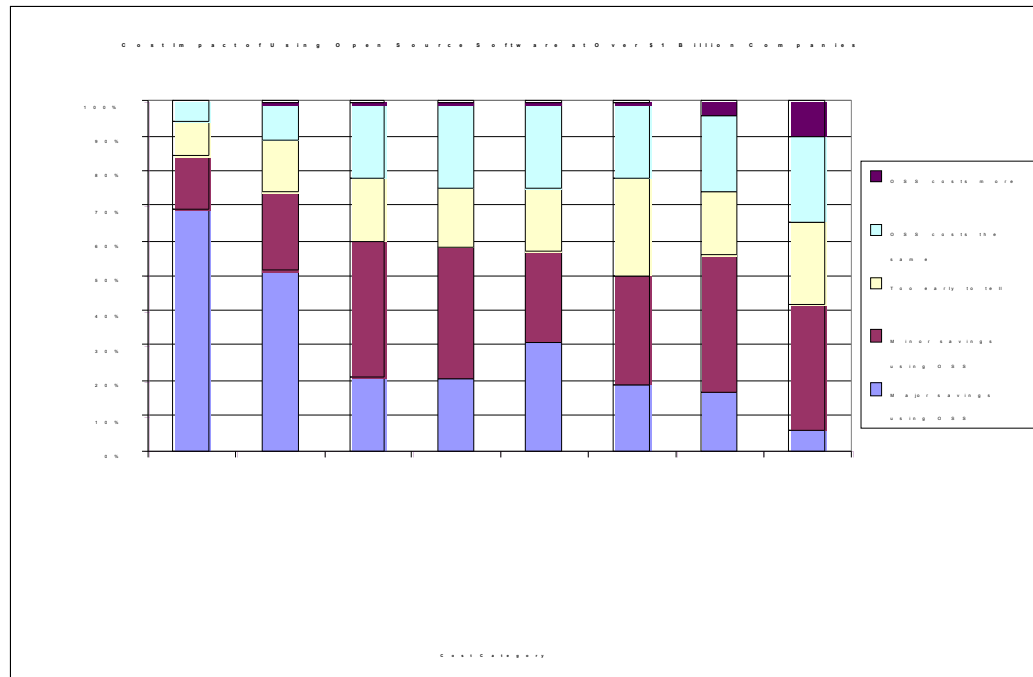
- Morgan Stanley
 - The ability to run any application, at any time, anywhere
 - Architecture decision to move to Linux enabled them to add 10K new servers and 100TB of storage without increasing headcount
- Orbitz
 - Providing a self-service architecture in the travel industry as it consolidates
 - Orbitz remains #1 in the industry based on customer ranking for the third year in a row
- Amazon
 - Uses IT to drive the revenue curve with such tools as customer reviews, the suggestion engine, and the virtual store across many categories

Value Versus Cost on the Income Statement

- Mainstream businesses invest 4%-8% in IT; Financial Services a little more (8%-12%) and Healthcare a little less (2%-4%)
- A significant cost savings on the IT budget will actually have little impact on a company's value compared to normal volatility in the market: Saving money on license fees is not a competitive strategy
- A significant revenue gain (10%) can have substantial impact on a company's value:
 - Improved customer retention
 - Market expansion (both market share and new market)
 - Time to market
 - Price competition

Cost Benefits of Open Source Software

Company Size	Average 2004 Open Source Cost Savings
Large (over \$1 billion)	\$3.3 million
Midsize (between \$ 50 million and \$1 billion)	\$1.1 million
Small (under \$50 million)	\$520,000



Once organizations begin using open source software, their usage increases.

Source: Optaros/InformationWeek Survey 2005, <http://www.optaros.com>

Copyright, Stephen R. Walli, some rights reserved

Distributed under a Creative Commons, Attribution-NonCommercial-NoDerivs 2.0 Deed

It's Just Software Licensing

- There's nothing inherently different about FOSS licensing and other software licensing
- FOSS licensing depends upon strong copyright laws
- “Dual” licensing is an attribute of IP law: you can license your property to as many people as many times and as many ways as you choose
- Collected works also have licenses
- It's all just software licensing – there is nothing particular (or peculiar) to open source here

In Summary

- It's just software -- but open source collaborative development is proving to be the best re-use strategy for vendors and customers alike
- It's just economics -- to “build” versus “buy”, we can again add “share” and “borrow”
- It's just business -- the value discussion with your collective vendors hasn't changed, but the items you're buying have

Questions?

- Email me: stephen.walli@gmail.com
- “Once more unto the Breach” -- a blog about open source, standards, and the business of software
<http://stephesblog.blogspot.com>